Supplement

1 Imputing missing data

Let M be the set of ordered pairs that are missing in the input matrix $\mathbf{T}$. Let $\mathbf{T}_{\mathsf{M}}$ be the cells of $\mathbf{T}$ that are missing, while $\mathbf{T}_{\mathsf{M}^c}$ represent the cells which are observed. As part of the E-step, we can impute the missing data by modelling all the pairs in M as latent variables, in addition to existing variables $\mathbf{W}$. We constrain $\mathrm{Q}(\mathbf{W}, \mathbf{T}_{\mathsf{M}})$ to factorize to $\mathrm{Q}(\mathbf{W})\mathrm{Q}(\mathbf{T}_{\mathsf{M}})$.

For inference on $\mathbf{T}_{\mathsf{M}}$, we solve an equivalent form of equation we did for $\mathbf{W}$. Define: $\mathrm{H}(\mathrm{Q}(\mathbf{T}_{\mathsf{M}})) = \mathbb{E}_{\mathrm{Q}(\mathbf{W})}\left[\mathrm{P}(\mathbf{W}, \mathbf{T}_{\mathsf{M}}|\mathbf{T}_{\mathsf{M}^c}; \mathbf{X}, \sigma^2)\right]$. From standard properties of the KL-Divergence,

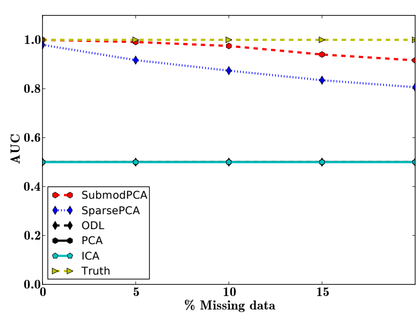
$$\begin{aligned} &\min_{\mathrm{Q}(\mathbf{T}_{\mathsf{M}})} \mathrm{KL}\big(\mathrm{Q}(\mathbf{T}_{\mathsf{M}})\mathrm{Q}(\mathbf{W}) \| \mathrm{P}(\mathbf{W}, \mathbf{T}_{\mathsf{M}}|\mathbf{T}_{\mathsf{M}^c}; \mathbf{X}, \sigma^2)\big). \\ \equiv &\min_{\mathrm{Q}(\mathbf{T}_{\mathsf{M}})} \mathbb{E}_{\mathrm{Q}(\mathbf{T}_{\mathsf{M}})}\left[\log \mathrm{Q}(\mathbf{T}_{\mathsf{M}})\right] - \mathbb{E}_{\mathrm{Q}(\mathbf{T}_{\mathsf{M}})}\left[\mathrm{H}(\mathrm{Q}(\mathbf{T}_{\mathsf{M}}))\right] \end{aligned} \tag{1}$$

Say $\mathrm{Q}(\mathbf{T}_{\mathsf{M}}) \sim \mathcal{N}(\mathbf{u}, \mathbf{\Gamma})$. Then Equation 1 is equivalent to

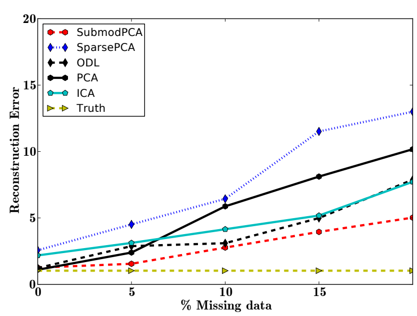
$$\min_{\mathbf{u},\mathbf{\Gamma}} -\log|\mathbf{\Gamma}| + \frac{1}{\sigma^2} \sum_{(i,j)\in\mathbf{T}_{\mathsf{M}^c}} (\mathbf{T}[i,j] - \mathbf{X}_{i,\cdot}^{\dagger}\mathbf{m}^i)^2 + \frac{\mathrm{tr}(\mathbf{\Gamma})}{\sigma^2} \tag{2}$$

Taking derivatives and setting to 0 results in: $\forall (i,j) \in \mathsf{M}, \mathbf{T}[i,j] \sim \mathcal{N}(\mathbf{X}_{i,\cdot}^{\dagger}\mathbf{W}_{\cdot,j}, \sigma^2\mathbf{I})$.

Thus, The E-step for $\mathbf{T}_{\mathsf{M}}$ results in: $\forall (i,j) \in \mathsf{M}, \mathbf{T}[i,j] \sim \mathcal{N}(\mathbf{X}_{i,\cdot}^{\dagger}\mathbf{W}_{\cdot,j}, \sigma^2\mathbf{I})$.



(a) Support Recovery as a function of missing data percent



(b) Reconstruction error as a function of missing data percent

Figure 1: Performance on Simulated Data

For missing data, we generate the data the same way as before. We randomly choose indices for which the data is not known, for various different percentages of missing data. Submodular Sparse PCA handles missing data inherently. However, the other methods have no support for the same. For comparison, the missing entries are filled with zeros for the other methods. Figure 1 presents the results for support recovery and reconstruction error on simulated data vis-a-vis baseline methods.

We present results for support recovery and reconstruction for different percentage of missing data. The results indicate that the method is robust to missing data, and degrades gracefully other methods as percentage of missing data increases. This can be explained by the fact that submodular sparse PCA inherently handles and imputes missing cells, while the other methods are operating with zeros filled in the empty cells.